Web Document Classification and its Performance Evaluation

IOAN POP
Department of Computer Science
"Lucian Blaga" University of Sibiu
5-7 Ioan Ratiu Str, Sibiu
ROMANIA

Abstract: The Web Mining applications have need to be improved with the specific algorithms for the document classification. This paper emphasizes the importance of using appropriate measures and methods for the evaluate of the Web document classification performance.We focus on methods that evaluate how well a classifier performs. The effect of transformations on the confusion matrix are considered for eleven well-known and recently introduced classification measures. We analyze the measure's ability to retain its value under changes in a confusion matrix. We discuss benefits from the use of the invariant and non-invariant measures with respect to characteristics of data classes.

Key-Words: document classification, performance, measure, confusion matrix

1. Introduction

1.1 Web Document Classification

The Web *document classification* means the assigning a document to a labeled predefine category. In the context related to informatic technologies applied in industries, business and economics in general, document classification is the process of the establishing a technical standard among competing entities in a market, where it will bring benefits without hurting competition. It can also be viewed as a mechanism for economic activity optimising.

The document classification tasks is divide in two categories: the *supervised document classification* where a extern tool (such as the human corecting reaction) provide the information of the corect classification for the documents and *unsupervised document classification* where the classification can be gived without to refer at extern information.

The work flow of the document classification process is illustrate in figure 1.

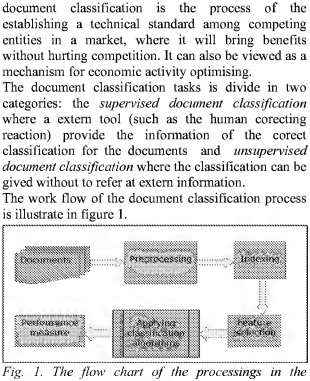

Fig. 1. The flow chart of the processings in the document classification process. [18]

Document classification process incudes the phases as follows:

- *Preprocessing*: transform documents into a suitable representation for classification task (i.e., remove HTML or other tags, remove stopwords, perform word stemming - remove suffix);
- *Indexing by different weighing schemes:* Boolean weighing, word frequency weighing, tf*idf weighing, ltc weighing, Entropy weighing, etc.;
- *Feature selection*: remove non-informative terms from documents that improve classification effectiveness and reduce computational complexity;
- *Classification algorithms*: Rocchio's algorithm, k-Nearest-Neighbor algorithm (KNN), Decision Tree algorithm (DT), Naive Bayes algorithm (NB), Artificial Neural Network algorithm (ANN), Support Vector Machine algorith (SVM), Voting algorithm, etc.;

Performance of algorithm: Training time, Testing time, Classification accuracy (precision, recall, F-score, micro-medie/macro- medie, etc.). The goal of this phase is a high classification quality and computation efficiency.

A *classifier* is a mapping from a (discrete or continuous) feature space X to a discrete set of labels Y. A framework of the classifier into Web document classification process is presented in figure 2.

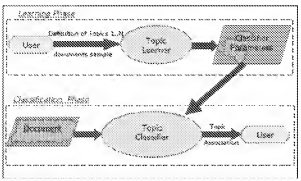

Fig. 2 A Framework of a classifier in two phases [19].

1.2 Techniques of the Web Document Classification

The architecture of a operational classification system in the automatic document classification is illustrated in figure 3.

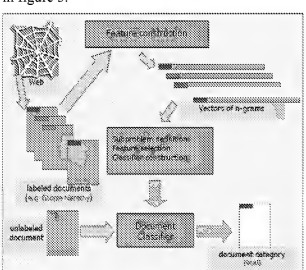

Figure 3. Operational system architecture of the document classification process. [19].

For the both task groups there are o vary document classification technics which improves allways The main category of the classification technics there are: Naive Bayes classifier, TF-IDF (**T**erm **F**requency – **I**nverse **D**ocument **F**requency), Latent Semantic Indexing (LSI), Support Vector Machine (SVM), Artificial Neural Network (ANN), k-Nearest Neighbor (kNN), Concept Mining, and approches based on natural language processing.

1.3 Web Mining and Web Document Classification

Web Mining is the extraction of interesting and potentially useful patterns and implicit information from artifacts or activity related to the World Wide Web. There are roughly three knowledge discovery domains that pertain to web mining: Web Content Mining, Web Structure Mining, and Web Usage Mining. In Figure 4 we illustrate the taxonomy of web mining [1].

Web content mining is the process of extracting knowledge from the content of documents or their descriptions. Web document text mining, resource discovery based on concepts indexing or agent-based technology may also fall in this category. Web structure mining is the process of inferring knowledge from the World Wide Web organization and links between references and referents in the Web. Finally, web usage mining, also known as Web Log Mining, is the process of extracting interesting patterns in web access logs.

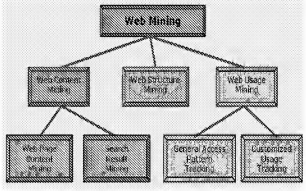

Fig. 4. Web Mining Taxonomy.

2 A Review of the Evaluation Metrics in classification

Most evaluation metrics in classification are designed to reward class uniformity in the example subsets induced by a feature (e.g., Information Gain). Other metrics are designed to reward discrimination power in the context of feature selection as a means to combat the feature-interaction problem (e.g., Relief, Contextual Merit).

2.1 Purity-based Metrics

An evaluation metric $\mathcal{M}$ quantifies the quality of the partitions induced by a feature X_k over a set of training examples T. Purity-based or traditional metrics define M by measuring the amount of class uniformity gained by decomposing T into the set of example subsets $\{T_m\}$ induced by X_k. Let $\vec{P}$ be the vector of class probabilities estimated from the data in the complete

set T, let $\vec{P}_m$ be the corresponding vector of class probabilities estimated from the data in the induced subset T_m, and let I be a measure of the impurity of a class probability vector. $\mathcal{M}$ is typically defined as follows [7]:

$$\mathcal{M}(X_k) = I(\vec{P}_m) - \frac{|T_m|}{|T|}\sum_m I(\vec{P}_m) \quad (1)$$

Different variations of $\mathcal{M}$ can be obtained by changing the impurity function I. For example, for Information Gain [8], impurity is defined in terms of entropy:

$$I_{entropy}(\vec{P}) = -\sum_i p_i \log_2 p_i \quad (2)$$

Another example is Gini Index [9]

$$I_{gini}(\vec{P}) = -\sum_i p_i^2 \quad (3)$$

Equation (1) covers most traditional metrics, but there are two major limitations:

- first is a tendency to favor features with many values. Inducing many example subsets increases the probability of finding class-uniform subsets, but at the expense of overfitting. Several solutions have already been proposed for this problem [3];
- second is the inability to detect the relevance of a feature when its contribution to the target concept is hidden by combinations with other features, also known as the feature-interaction problem [10].

To attack the feature-interaction problem additional information besides class probabilities is required.

2. 2 Discrimination-based Metrics

A different kind of evaluation metric considers the discrimination power of each feature, i.e., the ability of a feature to separate examples of different class. Let $\vec{X}_i$ and $\vec{X}_j$ be two examples lyingclose to each other according to some distance measure D. Feature X_k is awarded some amount of discrimination power if it takes on different values when the class values of $\vec{X}_i$ and $\vec{X}_j$ differ, i.e., when $x_k^i = x_k^j$ and $C(\vec{X}_i) \neq C(\vec{X}_j)$. The more often this condition is true for pairs of nearby examples, the higher the quality of feature X_k.

Two representative examples of discrination-based metrics are Contextual Merit and Relief [10]. Before describing them, we define the distance between to examples as follows:

$$D(\vec{X}_i, \vec{X}_j) = \sum_{k=1}^{n} d(x_k^i, x_k^j) \quad (4)$$

For nominal features $d(x_k^i, x_k^j)$ is defined as

$$d(x_k^i, x_k^j) = \begin{cases} 1 \ if \ x_k^i \neq x_k^j \\ 0 \ if \ x_k^i = x_k^j \end{cases} \quad (5)$$

For numeric features $d(x_k^i, x_k^j)$ is defined as

$$d(x_k^i, x_k^j) = \frac{|x_k^i - x_k^j|}{TH(x_k^i, x_k^j)} \quad (6)$$

where TH is a normalization factor, e.g., $MAX(X_k) - MIN(X_k)$ (difference between the maximum and minimum values observed for feature X_k in T).

Different metrics are obtained by varying the update function (4). The Relief algorithm, for example, updates score q_k as follows:

$$q_k = \begin{cases} q_k + d(x_k^i, x_k^j) \ if \ C(\vec{X}_i) \neq C(\vec{X}_j) \\ q_k - d(x_k^i, x_k^j) \ if \ C(\vec{X}_i) = C(\vec{X}_j) \end{cases} \quad (7)$$

Thus Relief updates q_k whenever the feature values of two neighbor examples differ; the score increases if their class values differ and decreases if they are the same. Contextual Merit updates q_k when both feature values and class values differ; it uses the update function:

$$q_k = q_k + \frac{d(x_k^i, x_k^j)}{D(\vec{X}_i, \vec{X}_j)} \ \ if \ \ C(\vec{X}_i) \neq C(\vec{X}_j) \quad (8)$$

The score of a feature decreases quadratically with the distance between two examples [11].

3. Performance Evaluation of the Web Document Classification

The performance of a classifier can be measured or estimated in a number of different ways. Which method to use is still subject to research and depends on the type of classification and type of data to be classified.

3.1 Web Document Classification and Performance Measures

Quality of classification can be assessed using a confusion matrix, i.e., records of correctly and incorrectly recognized examples for each class. Table 1 reports on binary classification [12].

		Predicted Class	
		Class=Yes	Class=No
Actual Class	Class=Yes	*tp*	*fn*
	Class=No	*fp*	*tn*

Table 1. A confusion matrix for binary classification

The confusion matrix includes: *tp* = true positive, *fn*= false negative, *fp* = false positive şi *tn* = true negative. The retrieval of relevant documents, or a positive class, is the most important task, thus focus is on *tp* classification. Importance of retrieval of positive examples is reflected by the choice of performance measures for text classification:

$$Precision = \frac{tp}{tp + fp} \tag{9}$$

$$Recall = \frac{tp}{tp + fn} \tag{10}$$

$$Fscore = \frac{\left(\beta^2 + 1\right)tp}{\left(\beta^2 + 1\right)tp + \beta^2 fn + fp} \tag{11}$$

$$BreakEvenPoint = \frac{tp}{tp + fp} = \frac{tp}{tp + fn} \tag{12}$$

Three measures evaluate the classifier performance by calculating the ratio of correctly classified positive examples to examples labeled as positives (*Precision*), positive examples in data (*Recall*), and total positive examples, labeled and from data, (*Fscore*). *BreakEvenPoint* essentially estimates when disagreement between data and algorithm labeling of positive examples is balanced (*fp* = *fn*). All these measures omit *tn* in their formulas, thus do not consider correct classification of negative examples.

In [14] Lee et al presents: the retrieval of a positive class, discrimination between classes, balance between retrieval of both classes are possible tasks whose importance depends on the problem at hand. So far, there is no common understanding on the choice of measures used to evaluate performance of classifiers in Web document. Employed performance measures are either

$$Accuracy = \frac{tp + tn}{tp + fn + fp + tn}, \tag{13}$$

wich is used in [14] and other works by this group, or *Precision, Recall, Fscore,* or corespondence between

$$Sensitivity = \frac{tp}{tp + fn} = Recall \tag{14}$$

and

$$Specificity = \frac{tn}{fp + tn} \tag{15}$$

reported in [16]. With different measures in use, it is important to know how performance evaluations, produced by those measures, relate to each other.

3.2 Invariance Properties of the Measures

Finding appropriate measure is possible by establishing *how comparable are the involved measures*. Following [17], we focus on the ability of a measure to preserve its value under a change in a confusion matrix. The invariance of a measure signals that it does not detect this change. Depending on the learning goals, non detection can be beneficial or adverse.

For instance, text classification extensively uses *Precision* and *Recall* (*Sensitivity*). These measures do not detect changes in *tn*, when all other matrix entries remain the same. In document classification, a large number of unrelated documents constitutes a negative class that lacks unifying characteristics (a multimodal negative class). The criterion for the performance of the classifier is its *performance on related documents* (a well-defined, unimodal, positive class) and may not depend on *tn*. *Precision* and *Recall* depend on *tp*, which shows agreement between data and algorithm labeling of positive examples, and *fp* and *fn*, which show disagreement between data and algorithm labeling of positive examples. Thus these measures provide the most important perspective on classifiers' performance for document classification. Another emerging application of text classification, classification of consumer reviews, works with highly related documents constituting unimodal positive and negative classes. Thus the evaluation measure may depend on classification of negative examples and reflect the *tn* change, when other matrix elements stay the same.

We examine the invariance properties with respect to basic changes of a matrix. Our claim is that the following invariance properties affect the measure's applicability and trustworthiness:

Exchange of *tp* with *tn* and *fn* with *fp* (t1) Table 2 shows the confusion matrix after the changes to the confusion matrix reported in Table 1. A measure is invariant if

$$m(tp; fn; tn; fp) = m(tn; fp; tp; fn) \tag{16}$$

This shows measure permanence with respect to classification results distribution. If the measure is invariant, then it does not distinguish *tp* from *tn* and *fn* from *fp* and may not recognize asymmetry of classification results. Thus it may not be trustworthy when classifiers are compared on data sets with different and/or unbalanced class distributions. For example, invariant measures may be more appropriate

for assessment of classification of consumer reviews then for document classification.

Change of true negative count (t2) Table 3 presents the resulting confusion matrix. A measure is invariant if

$$m(tp; fn; tn; fp) = m(tp; fn; tn'; fp) \quad (17)$$

This measure does not recognize specifying ability of classifiers. Such evaluation may be more applicable to domains with a multi-modal negative class, built as everything not positive".

	Predicted Class		
		Class=Yes	Class=No
Actual Class	Class=Yes	*tn*	*fp*
	Class=No	*fn*	*tp*

*Table 2. Confusion matrix after the exchange of **tp** with **tn** and **fn** with **fp**.*

	Predicted Class		
		Class=Yes	Class=No
Actual Class	Class=Yes	*tp*	*fn'*
	Class=No	*fp*	*tn*

Table 3. Confusion matrix after a change in true negative count.

If the measure is non-invariant, has t2, then it acknowledges ability of classifiers correctly identify negative examples. If the measure is able to do this, it may be reliable for comparison in domains with a well-defined, unimodal, negative class. In case of text classification, these invariant measures are suitable for evaluation of document classification and non-invariant measures are preferable for evaluation of such communications where criteria exist for positive as well as for negative results.

Change of a false count (t3) Table 4 reports the confusion matrix. A measure is invariant if

$$m(tp; fn; tn; fp) = m(tp; fn; tn; fp') \quad (18)$$

t3 indicates measure constancy if disagreement increases between the data and classifier labels. An invariant measure shows preference for data labels. In case of unreliable data labeling such measure may give misleading results.

	Predicted Class		
		Class=Yes	Class=No
Actual Class	Class=Yes	*tp*	*fn*
	Class=No	*fp'*	*tn*

Table 4. Confusion matrix after a change in false positive count.

A non-invariant measure may not be suitable for data with many counter examples. If classifier ranking improves when *fp* increases, the measure may favor a classifier prone to faux positives.

In case of t3, the use of invariant and non-invariant measures might be decided based on problem and data characteristics.

Classification scaling (t4) Table 5 presents the confusion matrix. A measure is invariant if

$$m(tp; fn; tn; fp) = m(k_1tp; k_2fn; k_2tn; k_1fp) \quad (19)$$

	Predicted Class		
		Class=Yes	Class=No
Actual Class	Class=Yes	k_1tp	k_2fn
	Class=No	k_1fp	k_2tn

Table 5. Confusion matrix after scaling.

This shows measure uniformity with respect to proportional changes of classification results. If the measure is non-invariant, then its applicability may depend on class sizes. If we expect that for different data sizes the same portion of examples exhibits positive (negative) characteristics, then the invariant measure may be a better choice for classifiers' evaluation. The non-invariant measures may be more reliable if we do not know how representative is the data sample in terms of proportion positive/negative examples (which is might be the case in web-posted consumer reviews).

4 Conclusion

Much work has been done in the research of classifier performance evaluation, comparison and classifier performance optimization, though the conclusion that can be drawn after conducting the literature survey is that most articles only focus on one optimization technique or one learning algorithm. Furthermore there are often discussions in the literature about which learning algorithm to use given a specific class of problem.

We have analyzed applicability of performance measures to different subfields of text classification. We have shown that document classification differs from classification of human communications, thus that these two types of text classification may require different set of performance measures. We have shown that the results of the classifier comparison depend on a number of factors, including invariant properties of the measures. We have considered effects of various transformations of the confusion matrix on several well-known performance measures. The invariance properties have lead to fine distinctions of relations between the measures and the data characteristics. One way to insure reliable evaluation is to employ a measure corresponding to the learning setting. The

next step would be to expand the list of connections between learning settings and evaluation measures.
This approach opens new directions for future work. First, we built a framework for the *two-dimensional* relations "measure vs invariance" and omitted decision theory relations. Note that the listed measures evaluate different decision aspects of the classifier performance.

References

[1] I. Pop, *WMHAS Model for Improvement Document Classification*, WSEAS International Conferences, Agios Nikolaos, Crete, Greece, July, 2007.

[2] T. M. Mitchell, *Machine Learning*, International Edition, McGraw-Hill Book Co, Singapore, ISBN 0-07-042807-7, (1997).

[3] I. H. Witten, E. Frank, *Data Mining: practical machine learning tools and techniques with Java implementations*, Academic Press, Morgan Kaufmann Publishers, ISBN: 1-55860-552-5, 1999.

[4] A. Andersson, P. Davidsson, and J. Lindén, *"Measure-based classifier performance evaluation", Pattern Recognition Letters*, volume: 20 issue: 11-13, North-Holland, Elsevier, 1999, pp. 1165-1173.

[5] N.T. van der Merwe, and A.J. Hoffman, *"Developing an efficient cross validation strategy to determine classifier performance (CVCP)"* in proceedings of International Joint Conference on Neural Networks, IJCNN '01, Volume: 3, 2001, pp. 1663-1668.

[6] G. Monari, and G. Dreyfus, *"Withdrawing an example from the training set: an analytical estimation of its effect on a non-linear parameterised model", Neurocomputing*, volume: 35 issue: 1-4, Elsevier, 2000, pp. 195-201.

[7] I. Pop, *The Use of the Predictive Standards in Web Mining Processing*, in the Proceedings of the 7-th European Conference - E-COMM-LINE 2006, September 18-19 Bucharest, ISBN 978-973-88046-0-9, 2006.

[8] J.R. Quinlan, "C4.5: Programs for Machine Learning", Morgaqn Kaufmann, 1994.

[9] L. Breiman, J.H. Olshen, R.A. Stone, *"Classification and Regression Trees"*, Wadsworth, Belmont, CA, 1984.

[10] I. Kononenko, *"On Biases in Estimating Multi-Valued Attributes"*, International Joint Conference on Artificial Intelligence, pp. 1034-1040, 1995.

[11] R. Vilalta, M. Brodie, D. Oblinger and I. Rish, *"A Unified Framework for Evaluation Metrics in Classification Using Decision Trees"*, Machine Learning: EMCL 2001: 12th European Conference on Machine Learning, Freiburg, Germany, September, 2001, Proceedings, Lecture Notes in Computer Science, Springer Berlin/Heidelberg, Vol. 2167/2001, pp. 503-511, 2001.

[12] C. Womser-Hacker, *„Theorie des Information Retrieval III: Evaluierung"*, in proceedings: *Grundlagen der praktischen Information und Dokumentation*. München. Saur, 5. Auflage 2004, ISBN 3-598-11675-6, ISBN 3-598-11674-8, 2004.

[13] M. Sokolova and G. Lapalme, *"Performance Measures in Classification of Human Communications"*, A study avalaible at http://www-etud.iro.umontreal.ca/~sokolovm/PerformanceMeasuresCamera.pdf, accessed in aug. 2007.

[14] J. Huang, C. X. Ling, *"Constructing New and Better Evaluation Measures for Machine Learning"*, avalaible at http://www.ijcai.org/papers07/Papers/IJCAI07-138.pdf, accessed in aug. 2007.

[15] B. Pang, L. Lee, S. Vaithyanathan, *"Thumbs up? sentiment classification using machine learning techniques"*, In: Proc Empirical Methods of Natural Language Processing EMNLP'02, pp. 79-86, 2002.

[16] M. Sokolova, *"Learning from Communication Data: Language in Electronic Business Negotiations"* Ph.D. dissertation, 2006.

[17] M. Sokolova, *"Assessing invariance properties of evaluation measures"*, In: Proceedings of the Workshop on Testing of Deployable Learning and Decision Systems, the 19th Neural Information Processing Systems Conference (NIPS 2006), 2006.

[18] J.-Y. Nie, *Clustering et classification des documents*, available at http://www.iro.umontreal.ca/~nie/IFT6255/Clustering.pdf, accesed 2007.

[19] G. Goller, J. Loning, T. Will, W. Wolff, *Automatic Document Classification: A thorough Evaluation of various Methods*, Intern. Symposiums für Informationswissenschaft, Darmstadt, Nov. 2000, pp. 145-162.

[20] D. Mladenic, *Solomon seminar*, Carnegie Mellon University and J.Stefan Institute, 6.6.2000 14. avalaible at www.quintelligence.com/gradiva/DataminingWeb.ppt, accessed may, 2007.